

LISTING OF THE CLAIMS:

1. (Currently Amended) A method of processing a text file in a computer application, comprising the steps:

forming a template ~~from~~ having literal fragments of the text file ~~that include literals for an~~
output text stream;

providing a macro class to map data from the text file to the computer application;

embedding in the template a position to the macro class; and

storing the template in a simple text format;

using the template as an overlay ~~for parsing incoming files~~ to parse the text file into
segments having data, or as a prototype to generate a segment of an output file;

~~using a macro class to map data from the text file to an application; and~~

~~embedding the macro class as a keyword within the template, wherein when the template~~
~~reaches the keyword, the template calls the macro class to further process the text file~~
~~according to a given set of rules.~~

said using step including the steps of:

i) reaching said pointer in said template, and

ii) when said pointer is reached, using said pointer to invoke said macro class and using said macro class to map data from one of the segments of the text file to the computer application.

2. (Original) A method according to Claim 1, wherein the macro class reads in a segment of the text file and uses the segment to initiate application update processing.
3. (Original) A method according to Claim 1, wherein the macro class derives data from the application and formats it into the text file.
4. (Original) A method according to Claim 1, wherein the macro class derives a template name from the invoking template and uses that name to invoke a next template to further process the text file.
5. (Original) A method according to Claim 1, further comprising the step of providing an interface controller to prevent structure clashes by placing text data into appropriate places in a complex object structure as the text file is processed.
6. (Currently Amended) A system for processing a text file in a computer application, comprising:

means for forming a template ~~from~~ having literal fragments of the text file ~~that include~~
~~literals for an output text stream;~~

means forming a macro class to map data from the text file to the computer application,
wherein a pointer to the macro class is embedded in the template;

~~means for storing the template in a simple text format;~~

means for using the template as an overlay ~~for parsing incoming files~~ to parse the text file
into segments having data, or as a prototype to generate a segment of an output file;

~~means for using a~~ said means for using including

i) means for using said pointer, when said pointer is reached in said template, to
invoke said macros class, and

ii) means to use said macro class to map data from one of the segments of the text file
to an application; and the computer application.

~~means for embedding the macro class as a keyword within the template, wherein when the~~
~~template reaches the keyword, the template calls the macro class to further process the text~~
~~file according to a given set of rules.~~

7. (Original) A system according to Claim 6, wherein the macro class reads in a segment of
the text file and uses the segment to initiate application update processing.

8. (Original) A system according to Claim 6, wherein the macro class derives data from the application and formats it into the text file.

9. (Original) A system according to Claim 6, further comprising an interface controller to prevent structure clashes by placing text data into appropriate places in a complex object structure as the text file is processed.

10. (Currently Amended) A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for processing a text file in a computer application, said method steps comprising:

forming a template ~~from~~ having literal fragments of the text file ~~that include literals for an output text stream;~~

providing a macros class to map data from the text file to the computer application;

embedding in the template a pointer to the means class; and

~~storing the template in a simple text format;~~

using the template as an overlay for parsing incoming files to parse the text file into segments having data, or as a prototype to generate a segment of an output file;

said using step including the steps of:

- i) reaching said pointer in said template, and
- ii) when said pointer is reached, using said pointer to invoke said macro class and using said macro class to map data from one of the segments of the text file to the computer application.

~~using a macro class to map data from the text file to an application; and~~

~~embedding the macro class as a keyword within the template, wherein when the template reaches the keyword, the template calls the macro class to further process the text file according to a given set of rules.~~

11. (Original) A program storage device according to Claim 10, wherein the macro class reads in a segment of the text file and uses the segment to initiate application update processing.

12. (Original) A program storage device according to Claim 10, wherein the macro class derives data from the application and formats it into the text file.

13. (Original) A program storage device according to Claim 10, wherein said method steps further comprise the step of providing an interface controller to prevent structure clashes by

placing text data into appropriate places in a complex object structure as the text file is processed.

14. (New) A method according to Claim 1, wherein:

the step of using said macro class includes the step of, said macro class using said name to invoke said another template to further process the text file.

15. (New) A system according to Claim 6, wherein:

when the macro class is invoked, a name for another template is passed to the macro class from said template; and

said macro class uses said name to invoke said another template to further process the text file;

16. (New) A program storage device according to Claim 10, wherein

when the macro class is invoked, a name for another template is passed to the macro class from said template; and

the step of using said macro class includes the steps of, said macro class using said means to invoke said another template to further process the text file.